**Image Project in Qunatela**

Open Source models has been converted first in onnx format. This format gives compatibility to convert model further based on hardware requirement

All above thing was done to improve models quantization. We were converting models 32 bits weight to 16 bits and it has been seen that models accuracy was not much hampered.

Different models used for study were

1. **SSRNET(Sparse Self-attention based Residual Network)** – Gender and age detection. It can be used for both framework tensorflow and pytorch

SSRNet it is a deep learning model primarly used for age estimation using facial images. It is desgined to predict the age of a person based on the features extracted from their face.

While SSRNet is primarily used for age estimation, it’s possible to adapt deep learning models like SSRNet for gender estimation as well. Gender estimation from facial images is a common task in computer vision and can be approached similarly to age estimation.

To adapt SSR-Net for gender estimation, you would typically need to make some modifications:

* **Dataset:** You would need a labeled dataset containing facial images annotated with gender labels (e.g., male or female).
* **Output Layer:** Modify the output layer of SSR-Net to predict gender instead of age. This might involve changing the number of output units and the activation function to suit the binary classification task (male or female).
* **Loss Function:** Choose an appropriate loss function for gender estimation. Binary cross-entropy loss is commonly used for binary classification tasks like gender estimation.
* **Training:** Retrain the modified SSR-Net model using the gender-labeled dataset. Fine-tuning the pre-trained weights on age estimation tasks might help the model learn gender-specific features more effectively.
* **Evaluation:** Evaluate the performance of the adapted SSR-Net model on a separate test dataset to assess its accuracy in gender estimation.

1. **MobileNet-SSD(Single shot multibox detector)** – it is a common algorithm used for object detection In images. It combines two components

* **MobileNet:** MobileNet is a lightweight convolutional neural network architecture designed for mobile and embedded vision applications. It's optimized for efficiency and speed, making it well-suited for deployment on devices with limited computational resources. MobileNet achieves this by using depthwise separable convolutions, which reduce the number of parameters and computational cost while maintaining reasonable accuracy.It serves as the feature extractor in mobileNetSSD part.
* **SSD (Single Shot Multibox Detector):** SSD is an object detection algorithm that directly predicts bounding boxes and class labels for multiple objects in a single pass through the network. It achieves this by simultaneously predicting multiple bounding boxes at different scales and aspect ratios using convolutional feature maps at different layers of the network. SSD is known for its speed and accuracy, making it suitable for real-time object detection applications.

By combining the MobileNet architecture with the SSD framework, MobileNet-SSD achieves a good balance between accuracy and efficiency, making it particularly useful for deployment on resource-constrained devices such as mobile phones, drones, and embedded systems. It's commonly used in various applications, including Object Detection in Images, Real-time Object Tracking.

1. **RFB(Receptive Field Block Net)net** – RFB-Net can be used for face detection by training it on a dataset that contains images with annotated bounding boxes around faces. During training, the model learns to detect faces by analyzing the features present in different regions of the input images.

Here's a simplified overview of how RFB-Net can be used for face detection:

1. **Data Preparation: Gather a dataset of images containing faces. Each image should be annotated with bounding boxes that specify the location and size of the faces.**
2. **Model Training: Train the RFB-Net model on the annotated dataset. During training, the model learns to detect faces by adjusting its internal parameters (weights) based on the input images and their corresponding annotations. The goal is to minimize the difference between the predicted bounding boxes and the ground truth annotations.**
3. **Inference: Once the model is trained, it can be used for face detection on new, unseen images. During inference, the trained model takes an input image and processes it through its layers to detect faces. The model outputs the coordinates of bounding boxes around detected faces, along with confidence scores indicating the likelihood that each bounding box contains a face.**
4. **Post-processing: After inference, post-processing techniques may be applied to refine the detected bounding boxes and filter out false positives. This could include techniques such as non-maximum suppression to merge overlapping bounding boxes or setting a threshold on confidence scores to remove low-confidence detections.**
5. **Evaluation: Evaluate the performance of the face detection model on a separate test dataset to assess its accuracy, precision, recall, and other metrics. Fine-tune the model and adjust hyperparameters as needed to improve performance.**

**In summary, RFB-Net is a neural network architecture designed to effectively capture information from a wide area of an input image, typically for object detection tasks. While it's not specifically designed only for face detection, it can be used for that purpose among others. Its ability to capture detailed information across a wide region of an image makes it well-suited for tasks where objects may vary in size and appearance, such as face detection in images with varying poses, lighting conditions, and backgrounds.**